

Precise vehicle ego-localization using feature matching of pavement images

Zijun Jiang

Chang'an University, Xi'an, China

Zhigang Xu

School of Information Engineering, Chang'an University, Xi'an, China, and

Yunchao Li, Haigen Min and Jingmei Zhou

Chang'an University, Xi'an, China

Abstract

Purpose – Precise vehicle localization is a basic and critical technique for various intelligent transportation system (ITS) applications. It also needs to adapt to the complex road environments in real-time. The global positioning system and the strap-down inertial navigation system are two common techniques in the field of vehicle localization. However, the localization accuracy, reliability and real-time performance of these two techniques can not satisfy the requirement of some critical ITS applications such as collision avoiding, vision enhancement and automatic parking. Aiming at the problems above, this paper aims to propose a precise vehicle ego-localization method based on image matching.

Design/methodology/approach – This study included three steps, Step 1, extraction of feature points. After getting the image, the local features in the pavement images were extracted using an improved speeded up robust features algorithm. Step 2, eliminate mismatch points. Using a random sample consensus algorithm to eliminate mismatched points of road image and make match point pairs more robust. Step 3, matching of feature points and trajectory generation.

Findings – Through the matching and validation of the extracted local feature points, the relative translation and rotation offsets between two consecutive pavement images were calculated, eventually, the trajectory of the vehicle was generated.

Originality/value – The experimental results show that the studied algorithm has an accuracy at decimeter-level and it fully meets the demand of the lane-level positioning in some critical ITS applications.

Keywords Feature extraction, Image matching, Intelligent transportation systems, Intelligent vehicles, Position measurement

Paper type Research paper

1. Introduction

Precise vehicle localization is one of the basic and urgent problems for most of the intelligent transportation system (ITS) applications. Through vehicle localization, many parameters associated with the working state of vehicles, such as vehicle position, velocity, acceleration and trajectory, can be obtained. These parameters are closely related to many security-themed applications in ITS. The literature (Boukerche *et al.*, 2008) lists over 10 applications closely related to the localization in ITS, which include routing navigation, data dissemination, map localization, adapted cruise control, cooperative intersection safety, blind crossing, platooning, vehicle collision warning, vision enhancement and automatic parking. It also points out that some applications such as vehicle collision warning, vision enhancement and automatic parking need sub-meter resolution. If precise localization information of all vehicles can be obtained in real-time, it will bring about revolutionary changes in future traffic

management. This can be specifically manifested in five aspects as follows:

- 1 In the event of a potential collision, for example, the potential risk for separated bicycle paths (Xu *C et al.*, 2016) and the merging vehicle's rear-end crash risk, an early warning can be made accurately (Weng *et al.*, 2015);
- 2 The accidents occurred in the past can be accurately reproduced through the recorded precise location data;
- 3 Some microscopic vehicle behaviors such as lane change, overtaking and motion in the wrong direction can be identified;
- 4 More timely and detailed road traffic situation can be obtained; and
- 5 Many new intelligent transportation applications can be invented through the accumulated precise vehicle

The current issue and full text archive of this journal is available on Emerald Insight at: <https://www.emerald.com/insight/2399-9802.htm>



Journal of Intelligent and Connected Vehicles
3/2 (2020) 37–47
Emerald Publishing Limited [ISSN 2399-9802]
[DOI 10.1108/JICV-12-2019-0015]

© Zijun Jiang, Zhigang Xu, Yunchao Li, Haigen Min and Jingmei Zhou. Published in *Journal of Intelligent and Connected Vehicles*. Published by Emerald Publishing Limited. This article is published under the Creative Commons Attribution (CC BY 4.0) licence. Anyone may reproduce, distribute, translate and create derivative works of this article (for both commercial and non-commercial purposes), subject to full attribution to the original publication and authors. The full terms of this licence may be seen at <http://creativecommons.org/licences/by/4.0/legalcode>

Received 22 December 2019

Revised 17 March 2020

Accepted 17 March 2020

trajectory data. In a word, if the real-time and precise localization problem were solved, it will speed up the step of “internet of vehicles” from theory to application. In this paper, the existing vehicle localization methods have been summarized through literature reviews.

1.1 Global navigation satellite system localization

GNSS stands for the global navigation satellite system, which refers to all satellite navigation systems, including global, regional and enhancements systems, such as the American global positioning system (GPS), the Russian GLONASS, the European Galileo, China’s BeiDou satellite navigation system and related enhancement systems, for example, the American wide-area augmentation system, Europe’s European Geostationary Navigation Overlay Service, Japan’s multi-functional transport satellite augmentation system, etc (Zaidi and Suddle, 2006).

GPS is currently the most common method of GNSS localization with the advantages of low cost, wide applications, all-weather working, etc. However, it also has limitations. On one hand, the satellite signals are blocked in the places of tunnels, mountain roads and city roads with skyscrapers surround, where GPS receiver cannot receive the satellite localization signals, on the other hand, the accuracy of GPS localization is typically between 20–30 m, which cannot achieve the accuracy at lane level. Aiming at the defects of the GPS, the differential GPS (DGPS) was designed as an improved method, which calculates pseudo-range correction in each satellite based on the reference station in the ground, and these corrections complete improvements in timing of the GPS satellite signals, track and atmosphere error. In general, the best localization accuracy of the DGPS is approximately 1 m. Unfortunately, when the number of satellites is seven or less due to buildings or trees, the average errors are more than a meter (Rezaii and Sengupta, 2007). The accuracy and reliability are still not high enough for some security applications such as collision warning, platooning and automatic parking.

1.2 Dead reckoning localization

The dead reckoning (DR) is a classic localization technology independent of the GNSS. For a moving object within a two-dimensional space, if its initial position and all the displacements at any previous time are known, the current position of the object can be calculated by the initial position added with accumulated displacement vectors, which relies on the inertial sensors such as odometer, gyroscopes, accelerometers and electronic compass to obtain the displacement and heading of a vehicle. The implementation of the DR system has two requirements as follows: the first is that the initial position of a moving target should be informed and the second is that the distance and direction of a moving target at all moments should be obtained (King et al., 2006). As the DR localization is an accumulation process, each estimated position of the target depends on the localization result of the previous moment. So, the measurement error and calculation error will accumulate with time elapsing, leading to a continuous decline in the DR localization. The DR system features high autonomy, high security, good resistance to radio interference, all-weather working, etc. Besides, it only uses its own inertial measurement components to deduce position,

speed and other navigation parameters. However, the accumulated errors of the DR system grow rapidly over time, so that it is unsuitable for long-term performance. Besides, it needs a long time for initial alignment, especially for position measurement (Bevly and Parkinson, 2007).

1.3 Integrated localization combined global positioning system with dead reckoning

As GPS and the DR are complementary, the localization precision can be improved by combining the two techniques. As an external input, the GPS information corrects the positioning result of DR frequently when the vehicle is in movement, which controls the accumulated error of DR as time going. On the contrary, the output of DR can solve the problems of the GPS in a short time, such as the loss of GPS signal and cycle slip in a complex environment, which strengthens the system’s anti-interference ability. The mutual penetration and combination of the two systems’ information can play a role in the complementary performance and improve the overall navigation precision and performance of the system (Krakiwsky et al., 1988). The overall performance after the combination of two systems is far better than each separate system, which becomes a hotspot in this research filed. There are several ways to fuse the localization information from multiple sensors in the GPS/DR integrated systems. Then, the fusion ways generally have three types, namely, non-coupling, tightly-coupling and loose-coupling. Among them, loose-coupling fusion has the best fault-tolerance performance, which uses local filter equations to fuse the output of the GPS and DR subsystems, and uses the main filter to fuse the output of the local filters. This approach not only reduces the system dimensions with the advantages of small calculation amount and parallel processing but also decreases the coupling degree of each subsystem. One sensor fault will not make a serious impact on other subsystems’ filter equations. The federal filter proposed by Carlson is a kind of loose-coupling fusion model, which attracts widespread attention because of its flexibility, small calculation amount and good fault-tolerance performance (Carlson, 1996).

1.4 Map matching localization

Map matching is a localization correction method based on software technology. Its basic idea is to associate a vehicle localization trajectory from a GPS receiver with the road information in an electronic map database, and thus, determine the vehicle position relative to the map (Chausse et al., 2005). The map matching applications are based on two premises. One is that all vehicles are always traveling on road; and the other is that the accuracy of the electronic map data should be higher than that of the estimated position of the road-vehicle navigation system. When the above conditions are met, the localization trajectory is compared with the road information through an appropriate matching process to determine the vehicle’s most likely traveling road section and its most likely position in this section. The map matching algorithm has a close relationship with the digital map (Jagadeesh et al., 2005). The electronic map must have the correct network topology and high accuracy to complete the map matching, otherwise, it will lead to false matches (Deusch et al., 2013).

1.5 Localization based image/video information

Images, videos and data processing techniques are often used for real-time localization in the field of autonomous vehicles and mobile robotics. These localization methods can be roughly divided into three categories as follows:

- 1 passive localization method based on video surveillance (Chapuis *et al.*, 2002);
- 2 ego-localization based on scene matching (Uchiyama *et al.*, 2009); and
- 3 ego-localization method based on visual odometry (VO) (Sakai *et al.*, 2010).

The video surveillance based passive localization tracks vehicle by cameras mounted on road infrastructures, which detects the target vehicle through background subtraction and calculates the vehicle's actual position through the camera calibration. However, it is difficult for the vehicle to obtain its own localization information from the video surveillance system. The localization method based on scene matching calculates the position of the vehicle by searching the most similar image in a pre-recorded image database or street view database with the captured images in real-time. The ego-localization method based on VO calculates the relative motion displacement between the two consecutive frames of the vehicle through matching overlapping areas among multi-frame images captured by the camera. Because of its relatively simple structure and high reliability, VO has been used in a wide variety of robotic applications such as on the Mars exploration rovers (Maimone *et al.*, 2007).

1.6 Radio localization

Radio localization is the process of finding the location of something through the use of radio waves. Generally, it first measures the transmission parameters of the radio waves, which travel from the known stationary objects to the moving target object such as the difference of time or phase and the variation of amplitude or frequency. From these parameters, the distance difference between the known objects and the target object, the moving direction of the target object can be calculated, which can be used to determine or predict the location of the moving target object (Sun *et al.*, 2005). One typical application of radio localization is the American 911 telephone system, which can acquire the localization of the person who dialing a mobile phone. In addition, there are other radio localization methods such as ultra-wideband, wireless-fidelity and cooperative localization method based on vehicular *ad hoc* network (Bahl and Padmanabhan, 2000; Lee and Scholtz, 2002; Cheng *et al.*, 2005; Thangavelu *et al.*, 2007). However, if this method is applied in vehicle localization, it requires a large number of roadside stations and needs high investment costs, which is clearly not suitable for vehicles' long-distance localization.

This paper presents a precise vehicle ego-localization method using feature matching of pavement images, which are captured by the camera installed at the rear of the car. On this basis, the local features of the pavement image are extracted based on speeded up robust features (SURF) descriptors, and the relative displacement and the rotation angle between two consecutive frames are obtained through the image matching. Finally, the trajectory of the vehicle is extracted and the precise

localization is achieved. The organization of this paper is as follows: in Section 2, we provide a simple literature review of the inertial navigation system (INS) assisted by the GPS and the INS assisted by the vision. Section 3 will introduce the experimental equipment and explain the flow of the entire algorithm. Section 4 illustrates a matching algorithm based on road image features. Section 5 describes the vehicle trajectory estimation algorithm. The experiment is described in Section 6, and Section 7 gives the conclusion.

2. Related work

Tao Wu and Ranganathan (2013) proposes a vehicle localization method using road markings. In his paper, the road markings (such as arrows, speed limits and zebras) were surveyed beforehand, and the corresponding GPS latitude and longitude are stored in the database. The pavement videos were captured by a color camera mounted on the roof of a car. With the developed detecting algorithms, the road markings were recognized and matched with those stored in the database. Once the road markings were matched successfully, the position of the vehicle can be calculated based on the stored GPS data. Tao Wu indicated that the proposed methods can achieve positioning accuracy at lane-level. It belongs to the global positioning method and requires surveying the positions of all road markings in advance, obviously not suitable for long-distance vehicle localization.

Chen proposed a perceptual fusion three-dimensional localization scheme for autonomous driving scenes using LIDAR and vision sensors, etc., efficiently generating three-dimensional candidate frames from a three-dimensional point cloud and combining features from multiple views divided by region get up and finish positioning (Chen *et al.*, 2017). Experiments show that this approach outperforms the state-of-the-art by around 25 and 30% AP on the tasks of three-dimensional localization and three-dimensional detection. In addition, for two-dimensional detection, this approach obtains 10.3% higher AP than the state-of-the-art on the hard data among the LIDAR-based methods. However, the price of LIDAR is relatively expensive, so LIDAR-based vehicle positioning methods can be difficult to implement.

Hiroyuki (Uchiyama *et al.*, 2009) from Nagoya University presents a vehicle ego-localization method using streetscape image sequences. The image sequences of two in-vehicle cameras are matched with a database that contains a sequence of streetscape images and their corresponding positions. A sequential image matching algorithm is developed to search for the image similarity with the captured image in the database. Eventually, the vehicle position is calculated based on triangulation using the positions stored in the database and the viewing directions of the two cameras. Based on experiments, the authors proved that the positioning accuracy of the proposed method is better than the GPS, and the horizontal positioning accuracy error is less than 1.5 m. This method requires establishing a huge database containing a large amount of streetscape images and it can hardly guarantee the system can work in real-time.

Vu *et al.* (2012) from the University of California, Riverside presents a sensor fusion technique that uses the computer vision and the differential pseudo-range DGPS measurements to aid the

INS. The proposed method mainly solve the localization problem in a challenging environment where the GPS signal is limited or unreliable. In Anh Vu's paper traffic lights were surveyed as landmarks and their location data is stored in a database in advance. The localization method uses satellite pseudo-range time-of-arrival measurements, Doppler measurements between satellites and the GPS antenna and previously mapped visual landmarks on an image taken by a camera that measures the angle-of-arrival to correct the INS. The experimental results have shown that the combination of the DGPS and a single visual feature measurement at 1 Hz is sufficient to achieve localization accuracy, which is typically less than 1 m. This method heavily relies on the aid of traffic lights, while there are almost no traffic lights on the expressway or rural roads.

Pink *et al.* (2009) propose a vehicle localization method based on aerial image matching. The method combines ideas from research on VO with a feature map that is automatically generated from aerial images into the visual navigation system. The presented method detects the road markings from the aerial images and the features of road markings are extracted to create a feature map. Two forward-looking cameras are fixed on the roof of a vehicle to capture the road images, an image processing algorithm is developed to match features from the cameras to previously generated feature map to obtain a precise vehicle localization result.

Turgay and Ahmed (2011) build a framework that uses stereo camera images and freely available satellite and road maps to automatically obtain accurate global vehicle localization. The forward pavement images are captured by two cameras on a car, the three-dimensional point cloud of the road surface is reconstructed based on the theory of stereoscopic. With the three-dimensional point cloud, the top-view images of the road are used to match with the satellite images. At first, the accurate vehicle poses, high-resolution top view images, map overlays and three-dimensional reconstructions of the road and its surroundings are all obtained.

In addition, Dean *et al.* (2008) propose a vehicle location method based on road terrain parameters including the road height change, the derivative of height and superelevation changes. Claus Brenner presents a vehicle localization method using landmarks obtained by the LIDAR mobile mapping system. Using associated landmark pairs and an estimation approach, the positions of the vehicle are obtained. From the literature listed above, we can find out most vision-based vehicle localization methods belong to the global positioning method, which needs to build a huge database previously and it is difficult to achieve real-time and long-distance localization of vehicles.

3. System setup and algorithm processing

This section describes the system setup and an overview of the proposed algorithm. The smart car uses the 2 million pixels Basler aca1600-60gc camera (60 frames/s, adjustable) on the campus of Chang'an University to capture the road image. Excessive vehicle speed will result in blurred pictures, which cannot detect and match the interesting points correctly. So, the vehicle speed is maintained in the range of 20-30 km/h, as shown in Figure 1. The data offline processing is implemented by MATLAB R2016a.

Figure 1 Smart car test platform



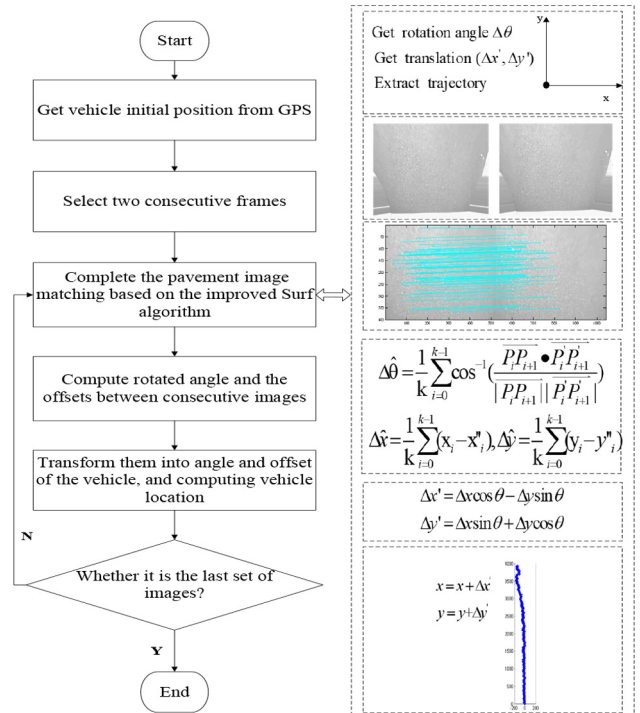
The general idea of this study is to achieve the precise vehicle localization using local feature matching of pavement images, as shown in Figure 2. First, the initial position of the vehicle is got by a GPS receiver. Second, we can get the top view of the pavement image. Then, we use the SURF operator to extract feature points of the two consecutive pavement images after correction to match feature points one by one and use a random sample consensus (RANSAC) algorithm to eliminate the false matching points. Finally, the relative translation and rotation offsets between the two consecutive images are computed with the selected matched points. With the known initial position and relative offsets between any two consecutive images, the vehicle's position can be obtained in real-time. This method is an ego-localization method, with relatively high robustness and precision. Furthermore, it is independent of landmarks, and there is no need to build up a database beforehand. The detail of the method was discussed as below.

4. Image matching

4.1 Comparison of the methods for local feature detection

This research mostly resolves the matching of the local features of pavement images, so it is very important to choose an

Figure 2 The overall flow chart of the algorithm



appropriate method to ensure the number of feature points and the efficiency of the algorithm. Different local features extraction algorithms are suited to the images with different features such as corners, blocks, spots and edges. The number of the detected feature points and the operation time are two key evaluation indicators for the local features extraction algorithms. Moreover, the size of the image is also a critical parameter, which directly determines the computation of an algorithm. To select an appropriate local features extraction algorithm and suitable image size, we conduct testing as follows:

- The originally captured pavement images are transferred into three different sizes, for example, $720 \times 1,280$ pixels, 360×640 pixels and 180×320 pixels shown in Figure 3;
- Four different corner detection algorithms (Harris, Susan, SIFT and SURF) are used to detect as candidates for detecting the feature points of the above images under the same computing environment. CPU: dual-core Intel 2.50 GHz, memory: 8G and platform: MATLAB R2016a; and
- An efficiency function is defined as follows:

$$P = \frac{\ln(N)}{T} \quad (1)$$

where N is the number of feature points, T is the processing time of the corresponding algorithm and P is the number of feature points matched in unit time.

The comparison results are shown in Table 1. Obviously, when the size of the image is equal to 360×640 pixels and the SURF is chosen as the local features extraction algorithm, the efficiency function P achieves the peak. So in this paper, we transfer all the original captured pavement images into 360×640 pixels and select SURF as the specified local feature extraction algorithm.

4.2 Detection and matching of the interesting points using improved speeded up robust features

According to Table 1, this study uses an improved SURF algorithm to detect the initial interesting points of pavement images. To further speed up the efficiency of feature points matching, this paper proposes a matching method based on the prejudgment of the dominant direction and the simplified distance formula. In addition, a simplified RANSAC algorithm is used to remove the false correspondence pairs. Finally, the robust matched feature points are acquired.

SURF is a good algorithm for the extraction and description of the local image features, which is primarily used in the field of image registration and stitching. The SURF algorithm includes three steps as follows:

- 1 the detection of the feature points;
- 2 the description of the feature points; and
- 3 the matching of the feature points.

The study in this paper optimized the SURF algorithm, a rapid and accurate matching algorithm is proposed, which makes the extraction of the vehicle trajectory more robust. Figure 4 shows the flow chart of the improved SURF algorithm.

4.2.1 Detection of the feature points

Feature points detection includes three steps as follows: the establishment of the integral image, the construction of the multi-scale space for the specified image using a box-type filter and the localization of feature points.

The rule judges whether a point (x, y) is a feature point or not can be described as follows:

- For a given threshold, if the determinant of the Hessian matrix of one pixel is greater than the threshold, it will turn to Step 2, else turn to next pixel;
- The non-maximum suppression is applied for $3 \times 3 \times 3$ three-dimensional neighborhood of the point, only the point, which is greater than all 26 response values in the three-dimensional neighborhood can be adopted as the candidate feature point; and
- To get a stable position and scale value of a candidate feature point, it needs to carry out interpolation operation on different scale space.

4.2.2 Description of the feature points

The feature described can be divided into two steps as follows: first, the dominant direction of the feature point is calculated to ensure the rotation invariance of the algorithm; second, the neighborhood of the feature points is rotated to the dominant direction, and the descriptor of the feature point is gained.

After the dominant direction of the feature point is determined, SURF uses wavelet responses in the horizontal and vertical direction to describe a distinctive feature point. A square region centered on the feature point and oriented along with its dominant orientation. The size of this window is 20×20 s, where s is the scale at which the feature point was detected. This square region is divided into 4×4 sub-regions with size 5×5 s. For each sub-region, a four-dimension feature vector is established as follows:

$$v = (\Sigma dx, \Sigma dy, \Sigma |dx|, \Sigma |dy|) \quad (2)$$

In equation (2), dx denotes the Haar wavelet response in a horizontal direction, and dy denotes the Haar wavelet response in a vertical direction. SURF also extracts the sum of the absolute values of the responses, $|dx|$ and $|dy|$ to enhance the robustness of the distinctive feature vector. Then, the vectors of 16 sub-regions are forming a 64 (4×16) dimension feature vector. To ensure its brightness and scale invariance, the descriptor must be normalized in advance.

4.2.3 Matching of the feature points

In this paper, there are three steps for feature point matching. First of all, fast index matching for preliminary screening of the SURF algorithm continues to be used. Second, the absolute distance is chosen to match the feature points and optimize the result of fast index matching. Third, the angle difference of the

Figure 3 Three different sizes of a pavement image

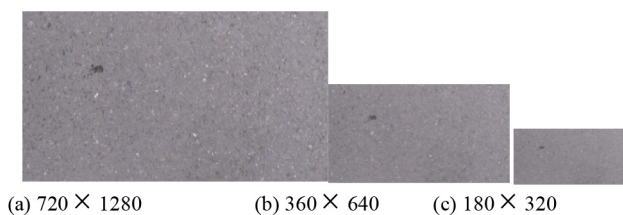
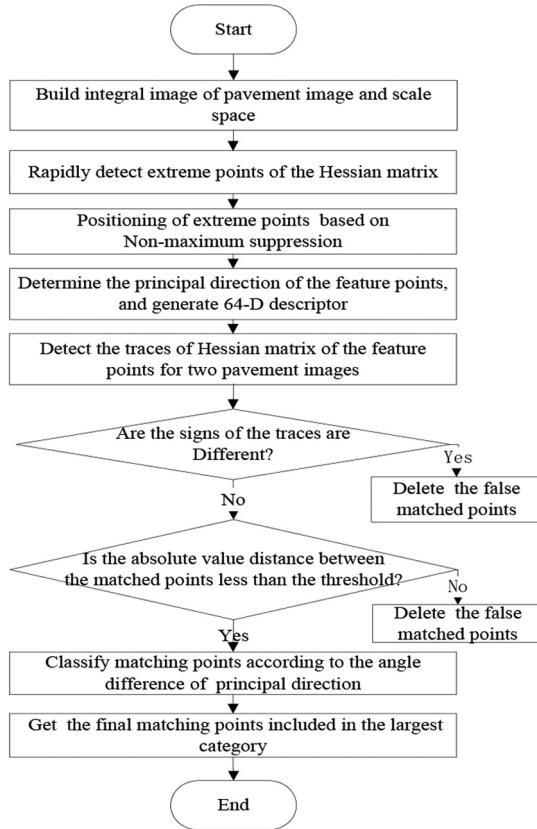


Table 1 Comparison of four kinds of algorithms

Image size	720 × 1,280	360 × 640	180 × 320
Detection time of Harris (s)	12.547	3.095	0.811
Corner number of Harris (n)	2,917	581	62
P	0.636	2.056	5.089
Detection time of SUSAN (s)	32.684	8.334	2.167
Corner number of SUSAN (n)	4,485	2,270	437
P	0.257	0.927	2.806
Detection time of SIFT (s)	254.863	48.528	10.641
Corner number of SIFT (n)	53,590	10,349	2,270
P	0.043	0.191	0.7267
Detection time of SURF (s)	8.894	0.895	0.379
Corner number of SURF (n)	1,820	201	5
P	0.844	5.925	4.247

Figure 4 The flow chart of improved SURF algorithm



dominant direction is used to eliminate the false correspondence pairs. Eventually, the final correct matched feature points are obtained:

- Fast index matching

In the process of the feature points detecting, the Hessian matrix trace is calculated. If the traces of two feature points have the same signs, it means that these two feature points have the same contrast. Otherwise, it indicates that they have different contrast and there is no need to measure the similarity between the two feature points, which can reduce the matching time and computation cost.

- The similarity measurement for the matched feature points based on the absolute distance

To describe the similarity of two feature points in two images, respectively, the absolute distance is used to calculate as follows:

$$L = \sum_{k=1}^{64} |l_{ik} - l_{jk}| \quad i = 1, 2, \dots, N_1; j = 1, 2, \dots, N_2 \quad (3)$$

In equation (3), l_{ik} denotes the k -th element of the i -th SURF feature point of the previous image. l_{jk} denotes the k th elements of the j -th SURF feature points of the current image. N_1 is the number of SURF feature points in a previous image and N_2 is the number in the current image.

For each feature point in the previous image, its absolute distances to all feature points in the current image are calculated, which constructs a distance set. From the distance set, we can select the minimum distance and the second minimum distance to compare with a threshold T . When the second minimum distance is less than T , this feature point in the previous image will be detained, as its corresponding feature point is found in the current image. Otherwise, we will discard this feature point. The smaller threshold is set, the less correspondence pairs will be reserved, while the distinctiveness and robustness of these pairs are higher. The proposed absolute distance is useful to improve the efficiency of the algorithm and also shortens the computation time with the comparison of the Euclidean distance.

- Elimination of the false correspondence pairs based on the angle difference

Taking into account image rotation, there is a certain angle difference among dominant directions of the matched points. $F1$ is a feature point in the previous image, which corresponds to the dominant direction ω_1 . $F2$ is a feature point in the current image, which corresponds to the dominant direction ω_2 . The angle difference between the two dominant directions is shown in equation (4):

$$\Delta\varphi = \omega_1 - \omega_2 \quad (4)$$

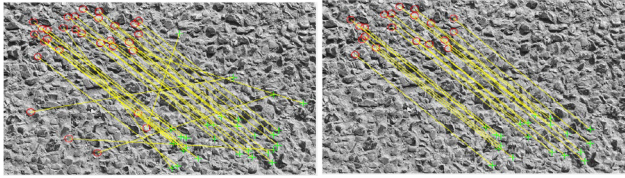
Image rotation reflects on the rotation of the feature points' dominant direction. If $\Delta\varphi$ is less than a threshold ($T1$), the

feature points can be reserved, else they will be eliminated as false matched points.

4.2.4 Elimination of the false matched points using random sample consensus

RANSAC algorithm is an iterative method to estimate parameters of a mathematical model from a set of observed data, which contains outliers. In this study, it is used to eliminate the false matched points. Every time, eight groups of points among all matched points are selected randomly to calculate the fundamental matrix, which can determine whether the rest of the points are inliers. The set with the maximum amount of inliers is considered as the final matched points set.

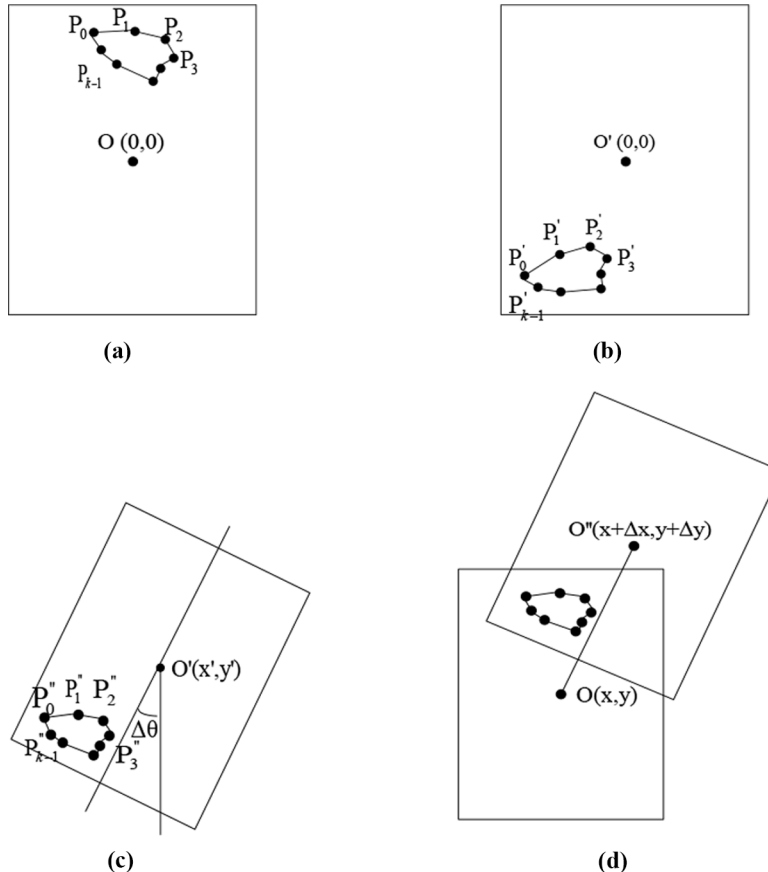
Figure 5 RANSAC to eliminate mismatched points



- The similarity distances of all correspondence pairs are sort in descending order;
- N ($N = \text{TotalNum} \times t$, TotalNum refers to the total number of matching points and t is a proportional factor) groups of correspondence pairs with bigger similarity distances are selected as the initial sample space;
- The fundamental matrix is calculated using the eight groups correspondence pairs randomly selected from the initial sample space and then the inliers can be detected according to the fundamental matrix; and
- Step 3 is repeated until the trial times come to the setting number. Finally, the points set having the maximum amount of inliers is considered as the one containing all the correct correspondence pairs.

In this study, the initial sample space is confined in the correspondence pairs, which have bigger similarity distances. So the fundamental matrix computed from this sample space would have higher compactness. As the sample space is narrowed, the computation time of the proposed RANSAC algorithm is reduced. On the other hand, it also decreases the total number of the final matched pairs. Considering the computation time and the total number of the remained correspondence pairs, in this study, the proportional factor t is set to 0.5-0.6 after experimental testing. The feature point

Figure 6 (a) The n -th image; (b) the $(n + 1)$ -th image; (c) the $(n + 1)$ -th image is rotated with an angle of $\Delta\theta$; and (d) the feature points in $(n + 1)$ -th image overlap with the ones in the n -th image by rotation and translation



matching diagram before and after using the RANSAC algorithm is shown in Figure 5.

5. Extraction of vehicle trajectory

Figures 6(a) and 6(b) are two consecutive images captured at time T_n and T_{n+1} , respectively. After completing the matching of the feature points in both images, the offset of vehicle movement during the sampling interval can be calculated through the coordinate transformation of the corresponding pairs. The feature points $P_0, P_1, P_2, P_3, \dots, P_{k-1}$ in the n -th image I_n and $P'_0, P'_1, P'_2, P'_3, \dots, P'_{k-1}$ in the $(n+1)$ -th image I_{n+1} represent the matched points set on the pavement in image coordinate system. When the vehicle moves, if the camera pose relative to the vehicle remains the same, the rigid transformation between the matched feature points in I_n and I_{n+1} can be described by equation (5):

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta & M \cdot \Delta x \\ \sin \theta & \cos \theta & M \cdot \Delta y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} \quad (5)$$

In this formula, (x', y') and (x, y) denote the coordinates of the feature points in I_{n+1} and I_n , respectively; $\Delta \theta$ denotes the rotation angle of the vehicle movement; Δx and Δy denote the horizontal and vertical offsets of the vehicle movement in the image coordinate system; M represents the scaling coefficient from the world coordinate to image coordinate. If I_{n+1} is rotated around its center O' with an angle of $\Delta \theta$ in counter-clockwise direction, and translated with an increment of Δx in horizontal and Δy vertical direction, these two images will completely overlap.

However, when the vehicle is moving in real-world coordination, the camera pose will constantly change due to vehicle vibration. Hence, the relationship between I_n and I_{n+1} becomes a projective transform instead of a strict rigid transformation, and the solution of equation (5) will be not a standard form shown in equation (6), which only includes the parameters regarding the rotation and translation. On the

contrary, it is a matrix form shown in equation (7), from which the rotation and movement offsets cannot be obtained directly:

$$H_r = \begin{bmatrix} \cos \theta & -\sin \theta & \Delta x \\ \sin \theta & \cos \theta & \Delta y \\ 0 & 0 & 1 \end{bmatrix} \quad (6)$$

$$H_p = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \quad (7)$$

In this paper, an approximate method is proposed to estimate the offset of vehicle movement. The key idea is to produce two arbitrary polygons with the same shape by connecting all the feature points in I_n and I_{n+1} ordered by the index. In Figure 6(c), after a rotation of $\Delta \theta$ and a translation of $(\Delta x, \Delta y)$, the polygon $P'_0 P'_1 P'_2 P'_3 \dots P'_{k-1}$ in I_{n+1} will approximately overlap with the polygon $P_0, P_1, P_2, P_3, \dots, P_{k-1}$ in Figure 6(d). The rotation angle $\Delta \theta$ can be estimated through equation (8) by averaging the included angles between the corresponding edges on the two polygons:

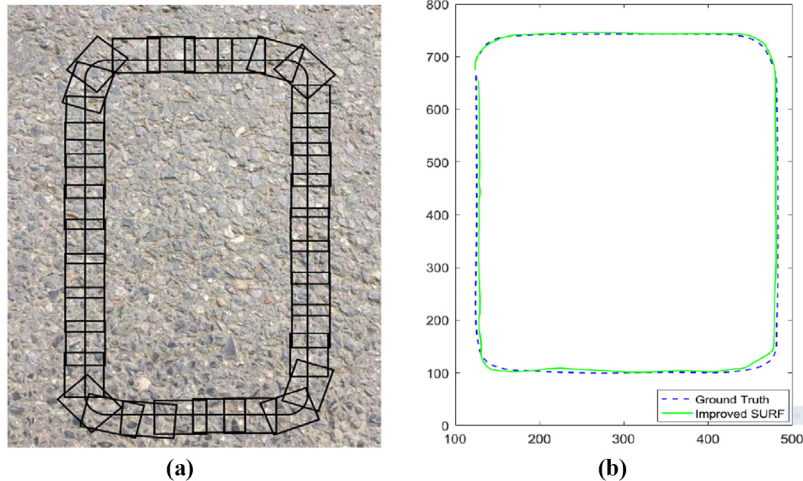
$$\Delta \hat{\theta} = \frac{1}{k} \sum_{i=0}^{k-1} \left(\arctan \frac{P'_{i+1}(x) - P_{i+1}(x)}{P'_{i+1}(y) - P_{i+1}(y)} - \arctan \frac{P'_i(x) - P_i(x)}{P'_i(y) - P_i(y)} \right) \quad (8)$$

In addition, Δx and Δy can be estimated by the translation of the gravity centers of the two polygons as shown in equation (9):

$$\begin{aligned} \Delta \hat{x} &= \frac{1}{k} \sum_{i=0}^{k-1} (x_i - x'_i) \\ \Delta \hat{y} &= \frac{1}{k} \sum_{i=0}^{k-1} (y_i - y'_i) \end{aligned} \quad (9)$$

As the offsets Δx and Δy are relative to coordinates XOY of the $(n+1)$ -th image, therefore, according to the rotated angle of two

Figure 7 Simulation trajectory



Notes: (a) Simulation trajectory rendering; (b) simulation by Matlab

images, offsets $\Delta x'$ and $\Delta y'$ can be calculated in the coordinate XOY of the n -th image, as defined by equation (10). If θ is the sum of the image rotated angles from the first image to the $(n+1)$ -th, $\Delta x'$ and $\Delta y'$ denote the offsets in the coordinate system of the first image. The average offsets of the $(n+1)$ -th image can be calculated based on all the final feature points:

$$\begin{cases} \Delta x' = \Delta x \cos \theta - \Delta y \sin \theta \\ \Delta y' = \Delta x \sin \theta + \Delta y \cos \theta \end{cases} \quad (10)$$

In this study, the GPS coordinate of the first image is taken as the initial position of the vehicle, and the above-average deviation is used to calculate the position corresponding to other images. The vehicle track can be plotted by connecting all the positions. Therefore, a track is drawn on an asphalt pavement as shown in Figure 7(a), and a track is obtained by using the above steps as shown in Figure 7(b). It can be seen that the two tracks basically coincide, thus verifying the feasibility of this algorithm.

6. Experimental results and analysis

To verify the correctness of the algorithm, we conducted a field experiment on the campus of Chang'an University, using the car shown in Figure 1 for data acquisition. In addition to the camera shown in the figure, the car was installed a DGPS system, and the DGPS system has a positioning accuracy of 2 m. Two experiments were carried out on the campus of Chang'an University. The first group of experiments is the road adaptability test; the second group is the open environment short-distance test of different maneuvering behaviors.

6.1 Road adaptability experiment

This topic selects three kinds of pavement images to test road adaptability. The three pavements are:

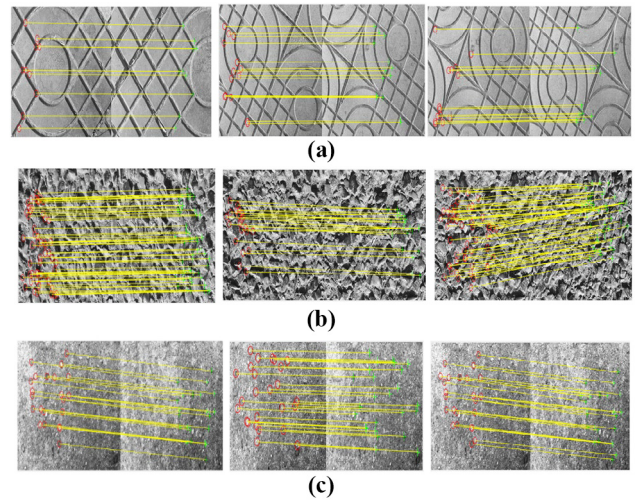
- 1 paving pavement;
- 2 asphalt pavement; and
- 3 cement pavement.

For each type of pavement, we collected 7,500 images for feature point acquisition and matching experiments. Figure 8 shows the SURF algorithm matching results for the three road surfaces. It can be seen from the results in the figure that for the image with a rich texture of the road surface, the SURF operator can get more matching feature points pairs because the surf operator is a multi-scale space feature point detection algorithm. Image spots at different scales are detected.

6.2 Short-distance experiment

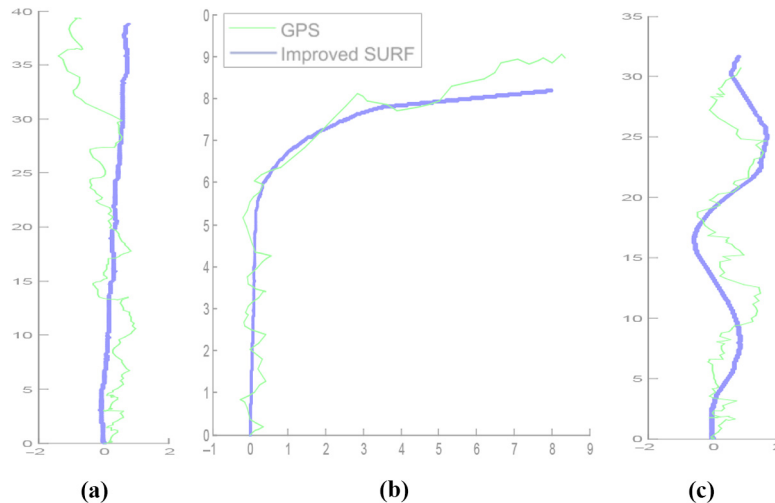
In our open environment, three different maneuvering trajectories were tested. The current vehicle speed is 30 km/h. The three maneuvers are straight, right and meandering. In this experiment, the DGPS positioning data is used as a reference. It

Figure 8 Surf algorithm matching results for three road surfaces



Notes: (a) Paving pavement; (b) asphalt pavement; (c) cement pavement

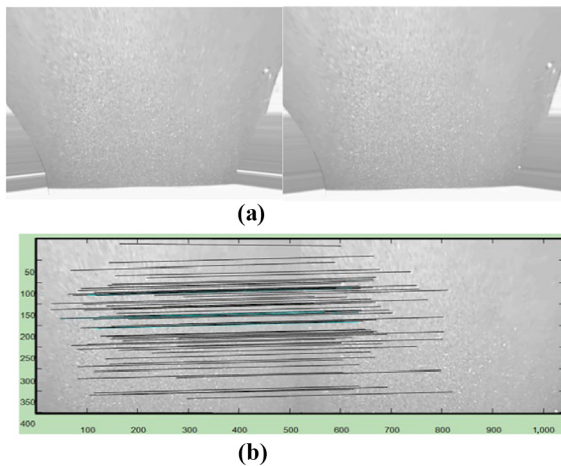
Figure 9 Short-distance experiment



Notes: (a) Straight trajectory; (b) turning trajectory; (c) lane changing trajectory

Table 2 The offsets of all matched pairs between two images

Offset pairs #	Offset				Corrected			
	Δx	Δy	$\Delta \theta$	$\Delta \theta$	Δx	Δy	$\Delta \theta$	$\Delta \theta$
1	-15.8614	-0.5742	/	/	47	-14.4780	0.6135	0.0250
2	-16.7136	-0.8217	-0.0114	-0.0114	48	-15.6444	0.0621	0.0059
3	-15.1938	0.7478	-0.0141	-0.0141	49	-16.8024	-0.5776	-0.0001
4	-15.5259	0.4458	0.0050	0.0050	50	-15.9476	0.5238	-0.0083
5	-15.7812	0.1316	-0.0045	-0.0045	51	-16.5887	-1.2866	-0.0042
6	-14.6177	0.9814	0.1881	0	52	-15.6603	-1.2565	0.0045
7	-16.1832	-0.3471	-0.0313	-0.0313	53	-15.8638	0.4115	0.0036
8	-16.1900	-0.5672	-0.0158	-0.0158	54	-15.5162	0.5364	-0.0017
9	-15.7571	0.1096	-0.0097	-0.0097	55	-15.0374	0.4458	0.0022
10	-14.0797	0.3885	-0.0564	-0.0564	56	-15.3338	0.0119	0.0026
11	-17.0772	-1.6135	-0.0425	-0.0425	57	-14.5871	0.7177	0.0153
12	-16.3072	-0.0515	-0.0237	-0.0237	58	-15.9907	-0.1268	0.0082
13	-16.2235	0.6755	-0.0049	-0.0049	59	-13.8391	-0.3000	0.0073
14	-15.0399	0.1378	3.1313	0	60	-15.7289	0.9493	-3.1131
15	-14.9716	1.1864	-0.0002	-0.0002	61	-15.8805	0.2761	-0.0005
16-46	Average	-15.6066	0.0851	/
								-0.0038

Figure 10 (a) The preprocessing results for 42-th and 43-th frame images; and (b) the matching results based on the improved SURF algorithm

can be seen from Figure 9 that the image positioning trajectory is smoother and the GPS data has a certain jitter. The detection accuracy is shown in the table. In the short distance case, the image positioning achieves the accuracy of the lane level.

Figure 10(a) shows the correction results of two consecutive images. Figure 10(b) shows the matched pairs between the two images, which are used to calculate the offsets and rotation angle between the two images. Table 2 shows the x -axis and y -axis offsets of 36 pairs of matching points, which are in Figure 10.

The average translation of all matched points is used as image offsets Δx and Δy . In addition, each pair of matching points is applied to calculate the rotated angle (Table 2) and the average of all the rotated angles is as the image rotated angle $\Delta \theta$. It may cause an error angle due to the error matching points, so the calculation results are amended by using the threshold of 0.05. The actual coordinates of the initial position are (a, b) and the logical coordinates are $(0, 0)$. According to equation (8), the image's offsets are calculated in the initial coordinates.

To test the accuracy of the trajectory, which is got by the algorithm in this study, three trajectories are compared to the data from GPS. The results are as shown in Figure 9. The analysis is shown in Table 3. It is easy to know that the vehicle trajectory accuracy of this experiment is better than that of GPS.

7. Conclusions

This study presents a precise vehicle ego-localization using local feature matching of pavement images, which captures pavement images by a reversing camera installed at the rear of a car. Through the matching of the two consecutive pavement images, the relative position of vehicle motion is obtained. This research can draw the following conclusions:

- The proposed method with low complexity and good real-time performance has an advantage that there is no need to establish a global database prior to use;
- After the comparison experiment, the pavement image feature extraction algorithm is suggested in this paper. Adopting the SURF for feature extraction, which can extract much more feature point with short processing time, it is very suitable for the pavement images;
- Experiment results show that the positioning accuracy of the new algorithm is less than 0.5 m, and it can satisfy the intelligent transportation application in the lane level;
- The studied algorithm is tested under daylight conditions, but for the night time conditions, the supplemental lighting equipment is also needed to enhance the overall image brightness; and
- The proposed algorithm may lead to undesired deviation in generating vehicle trajectory due to cumulative errors after a long run, so combining the GPS, the INS and other

Table 3 Comparison of vehicle location methods

Method	GPS	Proposed algorithm
Accuracy	<5 m	<0.5 m
Applications	Good signal area	Any region

positioning sensors together with the research is required to ensure the positioning with the long-term stability and reliability.

References

- Bahl, P. and Padmanabhan, V.N. (2000), "Radar: an in-building rf-based user location and tracking system", the 19th Annual Joint Conference of the IEEE Computer and Communications Societies, pp. 775-784.
- Bevly, D.M. and Parkinson, B. (2007), "Cascaded Kalman filters for accurate estimation of multiple biases, dead-reckoning navigation, and full state feedback control of ground vehicles", *IEEE Transactions on Control Systems Technology*, Vol. 15 No. 2, pp. 199-208.
- Boukerche, A., Oliveira, H.A.B.F. and Nakamura, E.F. (2008), "Vehicular ad hoc networks: a new challenge for localization-based systems", *Computer Communications*, Vol. 31 No. 12, pp. 2838-2849.
- Carlson, N.A. (1996), "Federated filter for computer-efficient, near-optimal GPS integration", *Proc: IEEE Position Location and Navigation Symposium*, pp. 306-314.
- Chapuis, R., Laneurit, J., and Aufrere, R. (2002), "Accurate vision based road tracker", *Proc: IEEE Intelligent Vehicle Symposium*, pp. 666-671.
- Chausse, F., Laneurit, J., and Chapuis, R. (2005), "Vehicle localization on a digital map using particles filtering", *Proc: IEEE Intelligent Vehicles Symposium*, pp. 243-248.
- Chen, X., Ma, H. and Wan, J. (2017), "Multi-view 3D object detection network for autonomous driving[C]", *IEEE Conference on Computer Vision and Pattern Recognition, Honolulu*, pp. 6526-6534.
- Cheng, Y.C., Chawathe, Y. and LaMarca, A. (2005), "Accuracy characterization for metropolitan-scale wi-fi localization", *Proc: the 3rd International Conference on Mobile Systems, Applications, and Services*, pp. 233-245.
- Dean, A.J., Martini, R.D. and Brennan, S.N. (2008), "Terrain-Based road vehicle localization using particle filters", *Proc: American Control Conference*, pp. 236-241.
- Deusch, H., Nuss, D. and Konrad, P. (2013), "Improving localization in digital maps with grid maps", 16th International IEEE Conference on Intelligent Transportation Systems, pp. 1522-1527.
- Jagadeesh, G.R., Srikanthan, T. and Zhang, X.D. (2005), "A map matching method for GPS based real-time vehicle location", *Journal of Navigation*, Vol. 57 No. 3, pp. 429-440.
- King, T., Füllner, H. and Transier, M. (2006), "Dead-Reckoning for position-based forwarding on highways", 3rd International Workshop on Intelligent Transportation, pp.199-204.
- Krakiwsky, E., Harris, C. and Wong, R. (1988), "A Kalman filter for integrating dead reckoning, map matching and GPS positioning", *Proc: Position Location and Navigation Symposium*, pp. 39-46.
- Lee, J.Y. and Scholtz, R. (2002), "Ranging in a dense multipath environment using an UWB radio link", *IEEE Journal on Selected Areas in Communications*, Vol. 20 No. 9, pp. 1677-1683.
- Maimone, M., Cheng, Y. and Matthies, L. (2007), "Two years of visual odometry on the Mars exploration rovers", *Journal of Field Robotics*, Vol. 24 No. 3, pp. 169-186.
- Pink, O., Moosmann, F., and Bachmann, A. (2009), "Visual features for vehicle localization and Ego-Motion estimation", *Proc: IEEE Intelligent Vehicle Symposium*, pp. 254-260.
- Rezaei, S. and Sengupta, R. (2007), "Kalman filter-based integration of DGPS and vehicle sensors for localization", *IEEE Transactions on Control Systems Technology*, Vol. 15 No. 6, pp. 1080-1088.
- Sakai, A., Tamura, Y. and Kuroda, Y. (2010), "Visual odometry using feature point and ground plane for urban environments", 41st International Symposium on Robotics, pp.1-8.
- Sun, G., Chen, J., Guo, W. and Liu, K. (2005), "Signal processing techniques in network-aided positioning: a survey of state-of-the-art positioning designs", *IEEE Signal Processing Magazine*, Vol. 22 No. 4, pp. 12-23.
- Thangavelu, A., Bhuvaneswari, K., Kumar, K., SenthilKumar, K. and Sivanandam, S. (2007), "Location identification and vehicle tracking using vanet", *Proc: International Conference on Signal Processing, Communications and Networking*, pp. 112-116.
- Turgay, Senlet and Ahmed, Elgammal (2011), "A framework for global vehicle localization using stereo images and satellite and road maps", *Proc: IEEE International Conference on Computer Vision Workshops*, pp.2034-2041.
- Uchiyama, H., Deguchi, D., and Takahashi, T. (2009), "Ego-localization using streetscape image sequences from in-vehicle cameras", *Proc: IEEE Intelligent Vehicle Symposium*, pp. 185-190.
- Vu, A., Ramanandan, A., Chen, A., Farrell, J.A. and Barth, M. (2012), "Real-Time computer vision/DGPS-Aided inertial navigation system for Lane-Level vehicle navigation", *IEEE Transactions on Intelligent Transportation Systems*, Vol. 13 No. 2, pp. 899-913.
- Weng, J., Xue, S., Yang, Y., Yan, X. and Qu, X. (2015), "In-depth analysis of drivers' merging behaviour and rear-end crash risks in work zone merging areas", *Accident Analysis & Prevention*, Vol. 77, pp. 51-61, doi: [10.1016/j.aap.2015.02.002](https://doi.org/10.1016/j.aap.2015.02.002).
- Wu, T. and Ranganathan, A. (2013), "Vehicle localization using road markings", *Proc: IEEE Intelligent Vehicle Symposium*, pp.1185-1190.
- Xu, C., Yang, Y., Jin, S., Qu, Z. and Hou, L. (2016), "Potential risk and its influencing factors for separated bicycle paths", *Accident Analysis and Prevention*, Vol. 87, pp. 59-67, doi: [10.1016/j.aap.2015.11.014](https://doi.org/10.1016/j.aap.2015.11.014).
- Zaidi, A.S. and Suddle, M.R. (2006), "Global navigation satellite systems: a survey", *International Conference on Advances in Space Technologies*, pp.84-87.

Corresponding author

Zhigang Xu can be contacted at: xuzhigang@chd.edu.cn

For instructions on how to order reprints of this article, please visit our website:

www.emeraldgroupublishing.com/licensing/reprints.htm

Or contact us for further details: permissions@emeraldinsight.com